AF
EFW

PTO/SB/21 (04-07)
Approved for use through 09/30/2007. OMB 0651-0031
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

TRANSMITTAL FORM

(to be used for all correspondence after initial filing)

Total Number of Pages in This Submission	

Application Number	09/760,795-Conf. #4941
Filing Date	January 17, 2001
First Named Inventor	Satoshi SETO
Art Unit	2157
Examiner Name	H. A. El Chanti
Attorney Docket Number	2091-0229P

ENCLOSURES (*Check all that apply*)

- [ ] Fee Transmittal Form
  - [ ] Fee Attached
- [ ] Amendment/Reply
  - [ ] After Final
  - [ ] Affidavits/declaration(s)
- [ ] Extension of Time Request
- [ ] Express Abandonment Request
- [ ] Information Disclosure Statement
- [ ] Certified Copy of Priority Document(s)
- [ ] Reply to Missing Parts/ Incomplete Application
  - [ ] Reply to Missing Parts under 37 CFR 1.52 or 1.53
- [ ] Drawing(s)
- [ ] Licensing-related Papers
- [ ] Petition
- [ ] Petition to Convert to a Provisional Application
- [ ] Power of Attorney, Revocation Change of Correspondence Address
- [ ] Terminal Disclaimer
- [ ] Request for Refund
- [ ] CD, Number of CD(s) _____
  - [ ] Landscape Table on CD
- [ ] After Allowance Communication to TC
- [ ] Appeal Communication to Board of Appeals and Interferences
- [ ] Appeal Communication to TC **(Appeal Notice, Brief, Reply Brief)**
- [ ] Proprietary Information
- [ ] Status Letter
- [X] Other Enclosure(s) (please Identify below):

  Reply to Notice of Non-Compliant Amendment

Remarks

SIGNATURE OF APPLICANT, ATTORNEY, OR AGENT

Firm Name	BIRCH, STEWART, KOLASCH & BIRCH, LLP		
Signature	Catherine M. Voorhees #52,327		
Printed name	Marc S. Weiner		
Date	August 20, 2007	Reg. No.	32,181

OIPE
AUG 20 2007
PATENT & TRADEMARK OFFICE

Docket No.: 2091-0229P
(PATENT)

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Patent Application of:
Satoshi SETO

Application No.: 09/760,795 | Confirmation No.: 4941

Filed: January 17, 2001 | Art Unit: 2157

For: IMAGE EDITING METHOD AND SYSTEM | Examiner: H. A. El Chanti

REPLY TO NOTICE OF NON-COMPLIANT APPEAL BRIEF

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

Responsive to the July 18, 2007 Notice of Non-Compliant Amendment, attached hereto is a corrected version of the "Appeal Brief" submitted on April 18, 2007.

The present submission is being timely filed within thirty (30) days of the mailing date of the Notice. Accordingly, no extension of time fees should be due.

If necessary, the Commissioner is hereby authorized in this, concurrent, and future replies, to charge payment or credit any overpayment to Deposit Account No. 02-2448 for any additional fees required under 37 C.F.R. §§ 1.16 or 1.17; particularly, extension of time fees.

Dated: August 20, 2007

Respectfully submitted,

By _____
Marc S. Weiner
Registration No.: 32,181
BIRCH, STEWART, KOLASCH & BIRCH, LLP
8110 Gatehouse Road
Suite 100 East
P.O. Box 747
Falls Church, Virginia 22040-0747
(703) 205-8000
Attorney for Applicant

OIPE
AUG 2 0 2007
PATENT & TRADEMARK OFFICE

MS APPEAL BRIEF
PATENT
2091-0229P

IN THE U.S. PATENT AND TRADEMARK OFFICE

In re Patent Application of:
Satoshi SETO

Application No.: 09/760,795 Confirmation No.: 4941

Filed: January 17, 2001 Art Unit: 2157

For: IMAGE EDITING METHOD AND SYSTEM Examiner: H. A. El Chanti

APPEAL BRIEF

TABLE OF CONTENTS

I. REAL PARTY IN INTEREST 1
II. RELATED APPEALS AND INTERFERENCES 2
III. STATUS OF THE CLAIMS 2
IV. STATUS OF AMENDMENTS 2
V. SUMMARY OF CLAIMED SUBJECT MATTER 2
VI. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL 11
VII. ARGUMENTS 11
VIII. CONCLUSION 37
IX. CLAIMS APPENDIX A-1
X. EVIDENCE APPENDIX B-1
XI. RELATED PROCEEDINGS APPENDIX C-1

OIPE AUG 20 2007 PATENT & TRADEMARK OFFICE

OIPE AUG 20 2007 PATENT & TRADEMARK OFFICE

MS APPEAL BRIEF
PATENT
0905-0236P

IN THE U.S. PATENT AND TRADEMARK OFFICE

In re Patent Application of:
Satoshi SETO

Application No.: 09/760,795 Confirmation No.: 4941

Filed: January 17, 2001 Art Unit: 2157

For: IMAGE EDITING METHOD AND SYSTEM Examiner: H. A. El Chanti

APPEAL BRIEF
ON BEHALF OF APPELLANT:
AKIHISA YAMAZAKI

MS APPEAL BRIEF
Board of Patent Appeals
and Interferences
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

Appellant hereby submits the following Appeal Brief in support of the Notice of Appeal filed October 17, 2006.

I. REAL PARTY IN INTEREST

The real party in interest is the assignee of the entire interest in the above-captioned patent application, FUJIFILM Corporation, 210 Nakanuma, Minami-Ashigara-shi, Kanagawa 250-0123, Japan.

II. RELATED APPEALS AND INTERFERENCES

There are no related appeals or interferences.

III. STATUS OF THE CLAIMS

Claims 1-52 are pending in the above-captioned application, and each of these claims is presently rejected. The rejection of claims 1-52 is being appealed.

IV. STATUS OF AMENDMENTS

Appellant filed amendments to the claims after the final rejection mailed April 18, 2006, however these amendments were not entered.

V. SUMMARY OF CLAIMED SUBJECT MATTER

The invention of claim 1 provides for an image editing method that is performed in an image editing system equipped with a client, which has an edit-command unit for applying a command to edit image data, and an image server, connected with said client through a network, which has an editing unit for obtaining processed image data by editing said image data in response to the edit command from said edit-command unit, said image editing method comprising a first step of accepting an edit-start command and, in response to said edit-start command, commanding said image server to transfer editing data, having at least one editing object, which contains said image data, at said edit-command unit, and of transferring said editing data to said client at said image server [Specification, page 31, line 21 through page 322, line 8]; a second step of querying said image server about one editing object for obtaining said processed image data in accordance with said editing data, at said edit-command unit [Specification, page 32, lines 5-7]; a third step of

transferring editing information, which represents said one editing object corresponding to said inquiry, to said client, at said editing unit [Specification, page 32, lines 8-11]; a fourth step of generating edit-command information which represents a command to edit said editing object, in accordance with said editing information and also transferring said edit-command information to said image server, at said edit-command unit [Specification, page 33, lines 2-4]; a fifth step of obtaining intermediate processed image data by applying an editing process to said editing data in accordance with said edit-command information and also transferring said intermediate processed image data to said client, at said editing unit [Specification, page 33, lines 11-15]; and a sixth step of repeating said second through the fifth steps, until said edit-command information is transferred for an editing object desired and said processed image data is obtained [Specification, page 33, line 23 through page 34, line 4].

The invention of claim 2 provides for an image editing system comprising a client having an edit-command unit for applying a command to edit image data [Specification, page 30, lines 7-9]; an image server 2, connected with said client through a network 3, which has an editing unit 7 for obtaining processed image data by editing said image data in response to the edit command from said edit-command unit [Specification, page 30, lines 15-22]; said edit-command unit having first means 10 for accepting an edit-start command and, in response to said edit-start command, commanding said image server to transfer editing data, having at least one editing object, which contains said image data [Specification, page 31, lines 14-21]; second means 10 for querying said image server about one editing object for obtaining said processed image data, based on said editing data transferred from said image server in accordance with said command to transfer said editing data [Specification, page 31, line 21 through page 32 line 2]; and third means 10 for generating edit-command information which represents a command to edit said editing object, based on said editing

information transferred from said image server in accordance with said inquiry about said editing object, and for transferring said edit-command information to said image server [Specification, page 32, lines 3-26]; said editing unit having first means 7 for transferring said editing data to said client in response to said command to transfer said editing data [Specification, page 31, lines 21-25]; second means 7 for transferring editing information, which represents an editing object corresponding to said inquiry, to said client [Specification, page 32, lines 8-11]; and third means 7 for obtaining intermediate processed image data by applying an editing process to said editing data, based on said edit-command information, and for transferring said intermediate processed image data to said client [Specification, page 33, lines 11-19]; and means 7 for repeatedly carrying out the steps carried out in the second and third means of said edit-command unit and the first, second, and third means of said editing unit, until said edit-command information is transferred for an editing object desired and said processed image data is obtained [Specification, page 33, lines 22-26].

The invention of claim 3 provides for a computer readable storage medium recording a program for causing a computer to carry out the image editing method as set forth in claim 1, wherein said program has a first procedure of accepting an edit-start command and, in response to said edit-start command, commanding said image server to transfer editing data, having at least one editing object, which contains said image data [Specification, page 31, lines 14-21]; a second procedure of querying said image server about one editing object for obtaining said processed image data, based on said editing data transferred from said image server in accordance with said command to transfer said editing data [Specification, page 31, line 21 through page 32 line 2]; a third procedure of generating edit-command information which represents a command to edit said editing object, based on said editing information transferred from said image server in accordance with said inquiry about said editing object, and of transferring said edit-command information to said image

server [Specification, page 32, lines 3-26]; and a fourth procedure of repeating said second and third procedures, until said edit-command information is transferred for an editing object desired and said processed image data is obtained [Specification, page 33, lines 22-26].

The invention of claim 4 discloses a computer readable storage medium recording a program for causing a computer to carry out the image editing method as set forth in claim 1, wherein said program has a first procedure of transferring said editing data to said client in response to said command to transfer said editing data [Specification, page 31, lines 21-25]; a second procedure of transferring editing information, which represents an editing object corresponding to said inquiry, to said client [Specification, page 32, lines 8-11]; a third procedure of obtaining intermediate processed image data by applying an editing process to said editing data, based on said edit-command information, and of transferring said intermediate processed image data to said client [Specification, page 33, lines 11-19]; and a fourth procedure of repeating said first, second, and third procedures, until said edit-command information is transferred for an editing object desired and said processed image data is obtained [Specification, page 33, lines 22-26].

The invention of claim 5 discloses an edit-command unit in an image editing system equipped with a client, which has said edit-command unit for applying a command to edit image data, and an image server, connected with said client through a network, which has an editing unit for obtaining processed image data by editing said image data in response to the edit command from said edit-command unit, said edit-command unit comprising first means 10 for accepting an edit-start command and, in response to said edit-start command, commanding said image server to transfer editing data, having at least one editing object, which contains said image data [Specification, page 31, lines 14-21]; second means 10 for querying said image server about one editing object for obtaining said processed image data, based on said editing data transferred from

said image server in accordance with said command to transfer said editing data [Specification, page 31, line 21 through page 32 line 2]; third means 10 for generating edit-command information which represents a command to edit said editing object, based on said editing information transferred from said image server in accordance with said inquiry about said editing object, and for transferring said edit-command information to said image server [Specification, page 32, lines 3-26]; and fourth means 10 for repeatedly carrying out the steps carried out in said second and third means, until said edit-command information is transferred for an editing object desired and said processed image data is obtained [Specification, page 33, lines 22-26].

The invention of claim 6 discloses an editing unit in an image editing system equipped with a client, which has an edit-command unit for giving a command to edit image data, and an image server, connected with said client through a network, which has said editing unit for obtaining processed image data by editing said image data in response to the edit command from said edit-command unit, said editing unit comprising first means 7 for transferring said editing data to said client in response to said command to transfer said editing data [Specification, page 31, lines 21-25]; second means 7 for transferring editing information, which represents an editing object corresponding to said inquiry, to said client [Specification, page 32, lines 8-11]; third means 7 for obtaining intermediate processed image data by applying an editing process to said editing data, based on said edit-command information, and for transferring said intermediate processed image data to said client [Specification, page 33, lines 11-19]; and fourth means 7 for repeatedly carrying out the steps carried out in said first, second, and third means, until said edit-command information is transferred for an editing object desired and said processed image data is obtained [Specification, page 33, lines 22-26].

The invention of claim 7 discloses an image editing method that is performed in an image editing system equipped with a client, which has an edit-command unit for giving a command to edit image data, and an image server, connected with said client through a network, which has an editing unit for obtaining processed image data by performing an editing process on said image data in response to the edit command from said edit-command unit and transfers predetermined image data related to said image data to said client, said image editing method comprising the steps of generating low-volume data smaller in data amount than said predetermined image data [Specification, page 38, lines 7-9]; and transferring said low-volume data to said client [Specification, page 38, lines 9-11].

The invention of claim 20 discloses an image editing system comprising a client having an edit-command unit for giving a command to edit image data [Specification, page 30, lines 7-9]; an image server, connected with said client through a network, which has an editing unit for obtaining processed image data by performing an editing process on said image data in response to the edit command from said edit-command unit and transfers predetermined image data related to said image data to said client [Specification, page 30, lines 15-22]; wherein said image server has means 9 for generating low-volume data smaller in data amount than said predetermined image data, and transfers said low-volume data to said client [Specification, page 38, lines 7-9].

The invention of claim 46 provides an image editing system comprising a client having an image-editing command unit for applying a command to edit image data representing a user's image [Specification, page 30, lines 7-9]; and a server, connected with said client through a network, which has means 5 for archiving said image data and low-resolution image data scaled down from said image data and edits said image data [Specification, page 37, lines 20-23]; editing information required for editing said image data which contains said low-resolution image data being transferred

from said server to said client [Specification, page 38, lines 2-12]; an operation of editing said low-resolution image data being performed at said client [Specification, page 38, lines 12 through page 39, line 7]; the result of editing being transferred to said server as edit-command information [Specification, page 39, lines 8-10]; processed image data being obtained by editing said image data according to said edit-command information at said server [Specification, page 39, lines 10-15]; wherein, when giving a command to insert a character image, which represents characters, into said user's image, said image-editing command unit generates character image data representing a character image of the same resolution as said user's image and transfers said character image data and said edit-command information to said server [Specification, page 41, line 18 through page 42, line 10]; and said image editing unit obtains said processed image data by inserting said character image into said user's image, based on said edit-command information and said character image data [Specification, page 42, lines 1-10].

The invention of claim 47 provides an image-editing command unit of an image editing system, equipped with a client having said image-editing command unit for applying a command to edit image data representing a user's image and a server which is connected with said client through a network and has means 5 for archiving said image data and low-resolution image data scaled down from said image data and edits said image data, in which editing information required for editing said image data which contains said low-resolution image data is transferred from said server to said client, an operation of editing said low-resolution image data is performed at said client, the result of editing is transferred to said server as edit-command information, and processed image data is obtained by editing said image data according to said edit-command information at said server, the image-editing command unit comprising means 10 which, when giving a command to insert a character image, which represents characters, into said user's image, generates character image data

representing a character image of the same resolution as said user's image and transfers said character image data and said edit-command information to said server [Specification, page 41, line 18 through page 42, line 10].

The invention of claim 49 provides an image-editing command method in an image editing system, equipped with a client having an image-editing command unit for applying a command to edit image data representing a user's image and a server which is connected with said client through a network and has means 5 for archiving said image data and low-resolution image data scaled down from said image data and edits said image data [Specification, page 27, lines 20-23], in which editing information required for editing said image data which contains said low-resolution image data is transferred from said server to said client [Specification, page 39, lines 10-15], an operation of editing said low-resolution image data is performed at said client, the result of editing is transferred to said server as edit-command information, and processed image data is obtained by editing said image data according to said edit-command information at said server[Specification, page 42, lines 1-10]; the image-editing command method comprising the steps of, when giving a command to insert a character image, which represents characters, into said user's image, generating character image data representing a character image of the same resolution as said user's image, and transferring said character image data and said edit-command information to said server [Specification, page 42, lines 1-10].

The invention of claim 51 provides, in a computer readable storage medium, recording a program for causing a computer to carry out an image-editing command method, in an image editing system, equipped with a client having an image-editing command unit for applying a command to edit image data representing a user's image and a server which is connected with said client through a network and has means 5 for archiving said image data and low-resolution image data scaled down

from said image data and edits said image data [Specification, page 27, lines 20-23], in which editing information required for editing said image data which contains said low-resolution image data is transferred from said server to said client [Specification, page 38, lines 2-12], an operation of editing said low-resolution image data is performed at said client, the result of editing is transferred to said server as edit-command information, and processed image data is obtained by editing said image data according to said edit-command information at said server [Specification, page 38, lines 12 through page 39, line 7], the computer readable storage medium wherein said program has the procedures of, when giving a command to insert a character image, which represents characters, into said user's image, generating character image data representing a character image of the same resolution as said user's image, and transferring said character image data and said edit-command information to said server [Specification, page 42, lines 1-10].

The summary of the claimed invention herein has been made to comply with the Patent Office rules in submitting briefs and is not to be considered as limiting the claimed invention.

VI. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

Claims 1-52 are rejected under 35 U.S.C. § 102(e) as being anticipated by *Fields et al.* (USP 6,412,008) (hereinafter, "*Fields*").

VII. ARGUMENTS

A. Argument Summary

The reasoning provided in support of the rejection of claims 1-52 under 35 U.S.C. § 102(e) as being anticipated by *Fields* fails to establish *prima facie* anticipation. Generally, the deficiencies of the rejection are that the rejection attributes certain claimed features to *Fields* that a detailed reading of the reference reveals are not taught therein. These deficiencies exist for the rejection of each of claims 1-52.

B. Legal Requirements of *Prima facie* Anticipation

In order to properly anticipate Appellant's claimed invention under 35 U.S.C. § 102(b), each and every element of the claim in issue must be found, either expressly described or under the principles of inherency, in a single prior art reference. *Verdegaal Bros. v. Union Oil Co. of California*, 814 F.2d 628, 631, 2 USPQ2d 1051 (Fed. Cir. 1987). "The identical invention must be shown in as complete detail as is contained in the . . . claims." *Richardson v. Suzuki Motor Co.*, 868 F.2d 1226, 1236, 9 USPQ2d 1913 (Fed. Cir. 1989). Finally, the elements must be arranged as required by the claims, but this is not an *ipsissimis verbis* test, i.e., identity of terminology is not required. *In re Bond*, 910 F.2d 831, 15 USPQ2d 1566 (Fed. Cir. 1990).

C. The Rejection Fails to Establish *Prima Facie* Anticipation of Independent Claim 1

Independent claim 1 is directed to an image editing method that is performed in an image editing system equipped with a client, which has an edit-command unit for applying a command to

edit image data, and an image server, connected with said client through a network, which has an editing unit for obtaining processed image data by editing said image data in response to the edit command from said edit-command unit, said image editing method comprising a first step of accepting an edit-start command and, in response to said edit-start command, commanding said image server to transfer editing data, having at least one editing object, which contains said image data, at said edit-command unit, and of transferring said editing data to said client at said image server; a second step of querying said image server about one editing object for obtaining said processed image data in accordance with said editing data, at said edit-command unit; a third step of transferring editing information, which represents said one editing object corresponding to said inquiry, to said client, at said editing unit; a fourth step of generating edit-command information which represents a command to edit said editing object, in accordance with said editing information and also transferring said edit-command information to said image server, at said edit-command unit; a fifth step of obtaining intermediate processed image data by applying an editing process to said editing data in accordance with said edit-command information and also transferring said intermediate processed image data to said client, at said editing unit; and a sixth step of repeating said second through the fifth steps, until said edit-command information is transferred for an editing object desired and said processed image data is obtained.

In support of his rejection of claim 1, the Examiner asserts in the final Official Action mailed April 18, 2006, on pages 2-3, as follows:

> As to claims 1 and 5, *Fields* teaches an image editing method that is performed in an image editing system equipped with a client, which has an edit-command unit for applying a command to edit image data, and an image server, connected with said client through a network, which has an editing unit for obtaining processed image data by editing said image data in response to the edit command from said edit-command unit, said image editing method comprising:
>
> a first step of accepting an edit-start command and, in response to said edit-start command, commanding said image server to transfer editing data, having at least one editing

object, which contains said image data, at said edit-command unit, and of transferring said editing data to said client at said image server (see col. 4 lines 50-64);

a second step of querying said image server about one editing object for obtaining said processed image data in accordance with said editing data, at said edit-command unit (see col. 4 lines 65-col. 5 lines 8);

a third step of transferring editing information, which represents said one editing object corresponding to said inquiry, to said client, at said editing unit (see col. 4, lines 65-col. 5 lines 8);

a fourth step of generating edit-command information which represents a command to edit said editing object, in accordance with said editing information and also transferring said edit-command information to said image server, at said edit-command unit (see col. 6 lines 10-30);

a fifth step of obtaining intermediate processed image data by applying an editing process to said editing data in accordance with said edit-command information and also transferring said intermediate processed image data to said client, at said editing unit (see col. 6 lines 32-45); and

a sixth step of repeating said second through the fifth steps, until said edit-command information is transferred for an editing object desired and said processed image data is obtained (col. 7 lines 1-57).

Appellant disagrees that *Fields* teaches a third step of transferring editing information, which represents said one editing object corresponding to said inquiry, to said client, at said editing unit. Further, there is no teaching or suggestion in *Fields* that is directed to a fourth step of generating edit-command information which represents a command to edit said editing object, in accordance with said editing information and also transferring said edit-command information to said image server, at said edit-command unit. Still further, there is no teaching or suggestion in *Fields* that is directed to a fifth step of obtaining intermediate processed image data by applying an editing process to said editing data in accordance with said edit-command information and also transferring said intermediate processed image data to said client, at said editing unit. Finally, there is no teaching or suggestion that is directed to a sixth step of repeating said second through fifth steps, until the edit-command information is transferred for an editing object desired and said processed image data is obtained.

1. ***Fields* fails to teach or suggest "a third step of transferring editing information, which represents said one editing object corresponding to said inquiry, to said client, at said editing unit" as required by independent claim 1**

The disclosure of *Fields* is directed to a system and method for cooperative client/server customization of web pages. *Fields* discloses a process by which a client sends a request for a network file, such as a web page, to a server. The server obtains the requested network file, and a server side customization program and customizes the file. The server side customization program may also analyze the network file and may embed return customization information in a customized network file. The client receives the customized network file, including the return customized information, from the server. A client side customization program then performs further customization on the network file (abstract).

The Examiner cites to col. 4, line 65 through col. 5, line 8 of *Fields* to support his rejection of the third step. At col. 4, line 65 through col. 5, line 8, *Fields* discloses as follows:

> Referring again to FIG. 2, server 32 obtains the requested network file, and server-side customization program 44 customizes the file according to the user agent string, the corporate options, and the personal options. Server-side customization program 44 also analyzes the network file, and may embed return customization information in customized file 60, as shown in FIGS. 3B and 3C. Return customization information 62 may be stored before file data 64 (as shown in FIG. 3B), or return customization information 62 may be embedded at various points in file data 64 (as shown in FIG. 3C).

As can be seen from this disclosure, *Fields* discloses one transfer of information, where the server obtains the requested network file, the server customizes the file, and embeds return customization information in the file.

In contrast, claim 1 clearly recites in the pre-amble a client having an edit-command unit. Claim 1 further clearly recites a first step of commanding the image server to transfer editing data, having at least one editing object, at the edit-command unit at the server, and transferring the editing data to the client at the image server (Step 1). In step 2 of claim 1, the image server is queried about the one editing object for obtaining the processed image data in accordance with the editing data (that was transferred from the image server to the client), at the edit-command unit at the client. At step 3 of claim 1, editing information is transferred, which represents the one editing object corresponding to the inquiry, to the client, at the editing unit.

These three steps clearly recite at least three transfers of information between the edit-command unit at the client and the image server.

However, as can be seen from Fig. 2 of *Fields*, there is only one request 50 submitted from the client to the server and one customized file 60 that is returned from the server to the client. There is no reasonable interpretation of the teachings of Fig. 2, which is more fully described in Figs. 5 and 6, and col. 4, line 30 through col. 6, line 45 that can teach or suggest the plurality of transfers of information that are recited in the claims in the first through third steps.

Further, there is no discussion in *Fields* that is directed to transferring editing information, which represents the one editing object corresponding to the inquiry, to the client at the editing unit. Claim 1 clearly recites that the editing data was transferred to the client, wherein the client queried the image server about the editing object and obtained processed image data in accordance with the editing data, where the editing information (representing the one editing object corresponding to the inquiry) was transferred to the client. There is no reasonable interpretation of the single exchange (one transfer of data from the client to the server, and one transfer of data from the server to the client) between the client and server of *Fields* that teaches these claim elements.

For at least these reasons set forth above, Appellant respectfully submits that the Examiner has failed to establish *prima facie* anticipation by failing to teach or suggest all of the claim elements. It is respectfully submitted that claim 1 is not anticipated by *Fields et al.*

2. *Fields* fails to teach or suggest "a fourth step of generating edit command information which represents a command to edit said editing object, in accordance with said editing information and also transferring said edit-command to said image server, at said edit command unit" as required by independent claim 1

The Examiner relies on col. 6, lines 10-30 in support of his assertion that *Fields* anticipates this claim element. At col.6, lines 1-45, *Fields* discloses as follows:

> Personal options may be set by a user via a graphical user interface, such as a pull-down menu, or via a Web page that saves a cookie with the personal options on the client's hard drive. Examples of some possible personal options are listed below. As noted above, the following list is not meant to be inclusive, and merely illustrates the wide range of personal options available:
> EXAMPLE PERSONAL OPTIONS

1. An option may be set to request a Web page in an alternate language, or to request that the Web page be translated into another language.
2. An option may be set to request a particular font or type size.
3. An option may be set to delete some or all of the graphics images or other multimedia elements on a Web page.
4. An option may be set to unroll or rearrange tables.
5. An option may be set to return a Web page as a list of links only.
6. An option may be set to color-code search terms for visibility (i.e. the Web page is being retrieved because it was a search engine result, and the user wishes the search terms to be highlighted on the resulting page).
7. An option may be set to crop images to a specific size, and display them with links to view the rest of the image.
8. An option may be set to format a file to a specific page width or length.

Referring again to FIG. 4, the client then sends an HTTP request to a server (step 102). The HTTP request includes the user agent string, the corporate options, and the personal options. The client then waits to receive the requested file from the server (step 104). The file that is received has been customized or modified according to the user agent string, and the requested corporate and personal options.

Next, the client checks to see if there is return customization information included in the returned file (step 106). If so, the client reads the return customization information (step 108). The client then performs further customization based on the return customization information, along with any other known user preferences and/or current conditions (step 110).

It is clear that the Examiner's reliance solely on lines 10-30 takes the teachings of *Fields* out context. As can be seen from the above citation, *Fields* discloses that the personal options may be set by a user. However, these settings are made prior to the initial communication request from the client to the server. Once the personal (and corporate) options are set, the client then initiates the exchange by sending an HTTP request to a server wherein the HTTP request includes the user agent string and the personal and corporate settings. All of this information is transferred in one request.

In contrast claim 1 recites a fourth step of generating edit command information which represents a command to edit said editing object, in accordance with said editing information and also transferring said edit-command to said image server, at said edit command unit.

It is clear from the citations set forth above that there is no teaching of suggestion in *Fields* that is directed to a fourth step of generating edit command information which represents a command to edit said editing object, in accordance with said editing information and also transferring said edit-command to said image server, at said edit command unit. *Fields* merely teaches the client device generating one request for a network file, where the request includes customization requests. There

is no teaching or suggestion in *Fields* that is directed to generating edit command information, represents a command to edit the editing object, wherein the editing object was transferred from the server to the client in step one. The request the Examiner is relying upon teaches, *arguendo*, commanding a server to transfer data to a client is recited, *inter alia,* in the first step of the claim. The Examiner appears to be relying on the same teachings of *Fields*, namely the request from the client to the server, as he was relying upon for the first step of the claim. This interpretation of the claim is wholly improper as the first step of the claim is different from the fourth step of the claim. There is no teaching in *Fields* that provides for a second command from the client for editing the editing object, as required by the fourth step of claim 1.

Further, there is no teaching or suggestion in *Fields* that is directed to generating edit command information representing a command to edit the editing object that was transferred from the server to the client in step 1. *Fields* only teaches one transmission from the client to the server.

For at least these reasons set forth above, Appellant respectfully submits that the Examiner has failed to establish *prima facie* anticipation by failing to teach or suggest all of the claim elements. It is respectfully submitted that claim 1 is not anticipated by *Fields et al.*

3. *Fields* fails to teach or suggest "a fifth step of obtaining intermediate processed image data by applying an editing process to said editing data in accordance with said edit-command information and also transferring said intermediate processed image data to said client, at said editing unit" as required by independent claim 1

The Examiner relies on col. 6, lines 32-45 in support of his assertion that *Fields* anticipates this claim element. The disclosure of *Fields* at col. 6, lines 32-45 is set forth above. However, as noted above, *Fields* fails to teach or suggest generating edit-command information, as recited in step 4 of claim 1. As such, *Fields* fails to teach or suggest obtaining intermediate processed image data by applying an editing process to the editing data in accordance with the edit commence information that was generated in step 4.

For at least these reasons set forth above, Appellant respectfully submits that the Examiner has failed to establish *prima facie* anticipation by failing to teach or suggest all of the claim elements. It is respectfully submitted that claim 1 is not anticipated by *Fields et al.*

4. *Fields* fails to teach or suggest "a sixth step of repeating said second through fifth steps, until said edit-command information is transferred for an editing object desired to said processed image data is obtained" as required by independent claim 1

The Examiner relies on col. 7, lines 1-57 in support of his assertion that *Fields* anticipates this claim element. The disclosure of *Fields* at col. 7, lines 1-57 merely describes the processes that take place at the server as described in Fig. 5, and an alternative embodiment where the client is a palmtop client, the process of which is described in Fig. 6. As can be seen in Figs. 5 and 6, again, only exchange of information is made between the client and server. As such, there is no teaching or suggestion that any of the steps recited in claim 1 is repeated.

For at least these reasons set forth above, Appellant respectfully submits that the Examiner has failed to establish *prima facie* anticipation by failing to teach or suggest all of the claim elements. It is respectfully submitted that claim 1 is not anticipated by *Fields et al.*

D. The Rejection Fails to Establish *Prima Facie* Anticipation of Independent Claim 2

Independent claim 2 is directed to an image editing system comprising a client having an edit-command unit for applying a command to edit image data; an image server, connected with said client through a network, which has an editing unit for obtaining processed image data by editing said image data in response to the edit command from said edit-command unit; said edit-command unit having first means for accepting an edit-start command and, in response to said edit-start command, commanding said image server to transfer editing data, having at least one editing object, which contains said image data; second means for querying said image server about one editing object for obtaining said processed image data, based on said editing data transferred from said image server in accordance with said command to transfer said editing data; and third means for generating edit-command information which represents a command to edit said editing object, based on said editing information transferred from said image server in accordance with said inquiry about said editing object, and for transferring said edit-command information to said image server; said editing unit having first means for transferring said editing data to said client in response to said command to transfer said editing data; second means for transferring editing information, which represents an editing object corresponding to said inquiry, to said client; and third means for

obtaining intermediate processed image data by applying an editing process to said editing data, based on said edit-command information, and for transferring said intermediate processed image data to said client; and means for repeatedly carrying out the steps carried out in the second and third means of said edit-command unit and the first, second, and third means of said editing unit, until said edit-command information is transferred for an editing object desired and said processed image data is obtained.

In support of the Examiner's rejection of claim 2, the Examiner asserts in the Final Official Action on pages 4-5, the Examiner asserts *Fields* anticipates the claim elements as follows:

> As to claim 2, *Fields* teaches an image editing system comprising:
> a client having an edit-command unit for applying a command to edit image data; an image server, connected with said client through a network, which has an editing unit for obtaining processed image data by editing said image data in response to the edit command from said edit-command unit (see col. 4 lines 50-col. 5 lines 8);
> said edit-command unit having first means for accepting an edit-start command and, in response to said edit-start command, commanding said image server to transfer editing data, having at least one editing object, which contains said image data; second means for querying said image server about one editing object for obtaining said processed image data, based on said editing data transferred from said image server in accordance with said command to transfer said editing data; and third means for generating edit-command information which represents a command to edit said editing object, based on said editing information transferred from said image server in accordance with said inquiry about said editing object, and for transferring said edit-command information to said image server (see col. 5 lines 10-col 6, lines 45);
> said editing unit having first means for transferring said editing data to said client in response to said command to transfer said editing data; second means for transferring editing information, which represents an editing object corresponding to said inquiry, to said client; and third means for obtaining intermediate processed image data by applying an editing process to said editing data, based on said edit-command information, and for transferring said intermediate processed image data to said client; and means for repeatedly carrying out the steps carried out in the second and third means of said edit-command unit and the first, second, and third means of said editing unit, until said edit-command information is transferred for an editing object desired and said processed image data is obtained (see col. 7 lines 1-57).

Appellant respectfully disagrees that *Fields* discloses the edit-command unit and the editing unit as recited in claim 2.

1. ***Fields* fails to teach or suggest "said edit-command unit having first means for accepting an edit-start command and, in response to said edit-start command, commanding said image server to transfer editing data, having at least one editing object, which contains said image data; second means for querying said image server about one editing object for obtaining said processed image data, based on said editing data transferred from said image server in accordance with said command to transfer said editing data; and third means for generating edit-command information which represents a command to edit said editing object, based on said editing information transferred from said image server in accordance with said inquiry about said editing object, and for transferring said edit-command information to said image server" as required by independent claim 2**

As noted above with regard to claim 1, *Fields* fails to teach or suggest querying said image server about one editing object for obtaining said processed image data, based on said editing data transferred from said image server in accordance with said command to transfer said editing data and further fails to teach or suggest for generating edit-command information which represents a command to edit said editing object, based on said editing information transferred from said image server in accordance with said inquiry about said editing object, and for transferring said edit-command information to said image server. As *Fields* fails to teach the functionality, there can be no teaching of means that performs the claimed functionality. As such, Appellant maintains that *Fields* fails to teach or suggest the edit-command unit performing the functionality as recited in claim 2. For at least these reasons, Appellant maintains that claim 2 is not anticipated by *Fields*.

2. ***Fields* fails to teach or suggest "said editing unit having first means for transferring said editing data to said client in response to said command to transfer said editing data; second means for transferring editing information, which represents an editing object corresponding to said inquiry, to said client; and third means for obtaining intermediate processed image data by applying an editing process to said editing data, based on said edit-command information, and for transferring said intermediate processed image data to said client; and means for repeatedly carrying out the steps carried out in the second and third means of said edit-command unit and the first, second, and third means of said editing unit, until said edit-command information is transferred for an editing object desired and said processed image data is obtained" as required by independent claim 2**

As noted above with regard to claim 1, *Fields* fails to teach or suggest transferring editing information, which represent an editing object corresponding to said inquiry, to said client. Further, as noted above, *Fields* fails to teach or suggest obtaining intermediate processed image data by applying an editing process to the editing data, based on the edit-command information and transferring the intermediate processed image data to the client. Still further, as noted above with regard to claim 1, *Fields* fails to teach or suggest repeatedly carrying out the steps until the edit-command information is transferred for an editing object desired and the processed image data is obtained. As *Fields* fails to teach the functionality, there can be no teaching of means that performs the claimed functionality. As such, Appellant maintains that *Fields* fails to teach or suggest the edit-command unit performing the functionality as recited in claim 2. For at least these reasons, Appellant maintains that claim 2 is not anticipated by *Fields*.

E. The Rejection Fails to Establish *Prima Facie* Anticipation of Independent Claim 5

Independent claim 5 is directed to an edit-command unit in an image editing system equipped with a client, which has said edit-command unit for applying a command to edit image data, and an

image server, connected with said client through a network, which has an editing unit for obtaining processed image data by editing said image data in response to the edit command from said edit-command unit, said edit-command unit comprising first means for accepting an edit-start command and, in response to said edit-start command, commanding said image server to transfer editing data, having at least one editing object, which contains said image data; second means for querying said image server about one editing object for obtaining said processed image data, based on said editing data transferred from said image server in accordance with said command to transfer said editing data; third means for generating edit-command information which represents a command to edit said editing object, based on said editing information transferred from said image server in accordance with said inquiry about said editing object, and for transferring said edit-command information to said image server; and fourth means for repeatedly carrying out the steps carried out in said second and third means, until said edit-command information is transferred for an editing object desired and said processed image data is obtained.

The Examiner's rejection of claim 5 is set forth above with regard to claim 1.

As noted above with regard to claim 1, Appellant maintains that *Fields* fails to teach or suggest querying said image server about one editing object for obtaining said processed image data, based on said editing data transferred from said image server in accordance with said command to transfer said editing data; generating edit-command information which represents a command to edit said editing object, based on said editing information transferred from said image server in accordance with said inquiry about said editing object, and for transferring said edit-command information to said image server; and repeatedly carrying out the steps until said edit-command information is transferred for an editing object desired and said processed image data is obtained.

As *Fields* fails to teach the functionality, there can be no teaching of means that performs the claimed functionality. As such, Appellant maintains that *Fields* fails to teach or suggest all of the elements recited in claim 5. For at least these reasons, Appellant maintains that claim 5 is not anticipated by *Fields*.

F. The Rejection Fails to Establish *Prima Facie* Anticipation of Independent Claim 6

Independent claim 6 is directed to an editing unit in an image editing system equipped with a client, which has an edit-command unit for giving a command to edit image data, and an image server, connected with said client through a network, which has said editing unit for obtaining processed image data by editing said image data in response to the edit command from said edit-command unit, said editing unit comprising first means for transferring said editing data to said client in response to said command to transfer said editing data; second means for transferring editing information, which represents an editing object corresponding to said inquiry, to said client; third means for obtaining intermediate processed image data by applying an editing process to said editing data, based on said edit-command information, and for transferring said intermediate processed image data to said client; and fourth means for repeatedly carrying out the steps carried out in said first, second, and third means, until said edit-command information is transferred for an editing object desired and said processed image data is obtained.

In support of the Examiner's rejection of claim 6, the Examiner asserts in the final Official Action on pages 6-7 as follows:

> As to claim 6, *Fields* teaches an editing unit in an image editing system equipped with a client, which has an edit-command unit for giving a command to edit image data, and an image server, connected with said client through a network, which has said editing unit for obtaining processed image data by editing said image data in response to the edit command from said edit-command unit, said editing unit comprising:

first means for transferring said editing data to said client in response to said command to transfer said editing data; second means for transferring editing information, which represents an editing object corresponding to said inquiry, to said client (see col. 4 lines 50-col. 5 lines 8);

third means for obtaining intermediate processed image data by applying an editing process to said editing data, based on said edit-command information, and for transferring said intermediate processed image data to said client (see col. 6 lines 10-54); and

fourth means for repeatedly carrying out the steps carried out in said first, second, and third means, until said edit-command information is transferred for an editing object desired and said processed image data is obtained (see col. 7 lines 1-54).

As noted above with regard to claim 1, Appellant maintains that *Fields* fails to teach or suggest obtaining intermediate processed image data by applying an editing process to said editing data, based on said edit-command information, and for transferring said intermediate processed image data to said client; and repeatedly carrying out the steps carried out in said first, second, and third means, until said edit-command information is transferred for an editing object desired and said processed image data is obtained.

As *Fields* fails to teach the functionality, there can be no teaching of means that performs the claimed functionality. As such, Appellant maintains that *Fields* fails to teach or suggest all of the elements recited in claim 6. For at least these reasons, Appellant maintains that claim 6 is not anticipated by *Fields*.

G. The Rejection Fails to Establish *Prima Facie* Anticipation of Independent Claim 7

Independent claim 7 is directed to an image editing method that is performed in an image editing system equipped with a client, which has an edit-command unit for giving a command to edit image data, and an image server, connected with said client through a network, which has an editing unit for obtaining processed image data by performing an editing process on said image data in response to the edit command from said edit-command unit and transfers predetermined image data

related to said image data to said client, said image editing method comprising the steps of generating low-volume data smaller in data amount than said predetermined image data; and transferring said low-volume data to said client.

In support of his rejection of claim 7, the Examiner asserts in the final Official Action on pages 7-8 as follows:

> As to claims 7 and 33, *Fields* teaches an image editing method and medium respectively that is performed in an image editing system equipped with a client which has an edit-command unit for giving a command to edit image data, and an image server, connected with said client through a network, which has an editing unit for obtaining processed image data by performing an editing process on said image data in response to the edit command from said edit-command unit and transfers predetermined image data related to said image data to said client, said image editing method comprising the steps of:
>
> generating low-volume data smaller in data amount than said predetermined image data; and transferring said low-volume data to said client (see col. 7 lines 1-54).

Appellants respectfully disagree that *Fields* teaches an image server, connected with said client through a network, which has an editing unit for obtaining processed image data by performing an editing process on said image data in response to the edit command from said edit-command unit and transfers predetermined image data related to said image data to said client, said image editing method comprising the steps of generating low-volume data smaller in data amount than said predetermined image data.

The disclosure of *Fields* merely describes a process whereby a server may eliminate graphics that are wider than the display size and modify table widths in order to fit them on the screen based on user agent strings and customization data.

However, claim 7 requires an image server which has an editing unit that transfers predetermined image data related to the image data to the client and generates low volume data smaller in data amount than the predetermined image data and transfers the low-volume data to the client. *Fields* fails to teach these elements as *Fields* only teaches transferring the customized data to the client. *Fields* fails to teach or suggest transferring both the predetermined image data and the low-volume data smaller in data amount than the predetermined image data to the client.

As *Fields* fails to teach or suggest all of the claim elements, Appellant maintains that claim 7 is not anticipated by *Fields*.

H. The Rejection Fails to Establish *Prima Facie* Anticipation of Independent Claim 20

Independent claim 20 is directed to an image editing system comprising a client having an edit-command unit for giving a command to edit image data; an image server, connected with said client through a network, which has an editing unit for obtaining processed image data by performing an editing process on said image data in response to the edit command from said edit-command unit and transfers predetermined image data related to said image data to said client; wherein said image server has means for generating low-volume data smaller in data amount than said predetermined image data, and transfers said low-volume data to said client.

In support of his rejection of claim 20, the Examiner fails to provide any statement in support of his rejection.

As noted above with regard to claim 7, *Fields* only teaches transferring the customized data to the client. *Fields* fails to teach or suggest transferring both the predetermined image data and the low-volume data smaller in data amount than the predetermined image data to the client. As *Fields* fails to teach the functionality, there can be no teaching of means that performs the claimed functionality. As such, Appellant maintains that *Fields* fails to teach or suggest all of the elements recited in claim 20. For at least these reasons, Appellant maintains that claim 20 is not anticipated by *Fields*.

I. The Rejection Fails to Establish *Prima Facie* Anticipation of Independent Claim 46

Independent claim 46 is directed to an image editing system comprising a client having an image-editing command unit for applying a command to edit image data representing a user's image; and a server, connected with said client through a network, which has means for archiving said image data and low-resolution image data scaled down from said image data and edits said image data; editing information required for editing said image data which contains said low-resolution image data being transferred from said server to said client; an operation of editing said low-resolution image data being performed at said client; the result of editing being transferred to said server as edit-command information; processed image data being obtained by editing said image data

according to said edit-command information at said server; wherein, when giving a command to insert a character image, which represents characters, into said user's image, said image-editing command unit generates character image data representing a character image of the same resolution as said user's image and transfers said character image data and said edit-command information to said server; and said image editing unit obtains said processed image data by inserting said character image into said user's image, based on said edit-command information and said character image data.

In support of his rejection of claim 46, the Examiner asserts in the final Official Action on page 10-11 as follows:

> As to claim 46 *Fields* teaches an image editing system comprising:
> a client having an image-editing command unit for applying a command to edit image data representing a user's image; and
> a server, connected with said client through a network, which has means for archiving said image data and low-resolution image data scaled down from said image data and edits said image data (see col. 6 lines 10-54);
> editing information required for editing said image data which contains said low-resolution image data being transferred from said server to said client; an operation of editing said low-resolution image data being performed at said client; the result of editing being transferred to said server as edit-command information (see col. 6 lines 10-54);
> processed image data being obtained by editing said image data according to said edit-command information at said server; wherein, when giving a command to insert a character image, which represents characters, into said user's image, said image-editing command unit generates character image data representing a character image of the same resolution as said user's image and transfers said character image data and said edit-command information to said server; and said image editing unit obtains said processed image data by inserting said character image into said user's image, based on said edit-command information and said character image data (see col. 7 lines 1-57).

Appellant respectfully disagrees that *Fields* discloses a server, connected with said client through a network, which has means for archiving said image data and low-resolution image data scaled down from said image data and edits said image data; editing information required for editing said image data which contains said low-resolution image data being transferred from said server to said client; an operation of editing said low-resolution image data being performed at said client; the result of editing being transferred to said server as edit-command information; processed image data being obtained by editing said image data according to said edit-command information at said server; wherein, when giving a command to insert a character image, which represents characters, into said user's image, said image-editing command unit generates character image data representing

a character image of the same resolution as said user's image and transfers said character image data and said edit-command information to said server; and said image editing unit obtains said processed image data by inserting said character image into said user's image, based on said edit-command information and said character image data.

1. ***Fields* fails to teach or suggest "a server, connected with said client through a network, which has means for archiving said image data and low-resolution image data scaled down from said image data and edits said image data" as required by independent claim 46**

The Examiner relies on col. 6, lines 10-54 to teach the above element. At col. 6, lines 10-54, *Fields* discloses as follows:

> EXAMPLE PERSONAL OPTIONS
> 1. An option may be set to request a Web page in an alternate language, or to request that the Web page be translated into another language.
>
> 2. An option may be set to request a particular font or type size.
>
> 3. An option may be set to delete some or all of the graphics images or other multimedia elements on a Web page.
>
> 4. An option may be set to unroll or rearrange tables.
>
> 5. An option may be set to return a Web page as a list of links only.
>
> 6. An option may be set to color-code search terms for visibility (i.e. the Web page is being retrieved because it was a search engine result, and the user wishes the search terms to be highlighted on the resulting page).
>
> 7. An option may be set to crop images to a specific size, and display them with links to view the rest of the image.
>
> 8. An option may be set to format a file to a specific page width or length.
>
> Referring again to FIG. 4, the client then sends an HTTP request to a server (step 102). The HTTP request includes the user agent string, the corporate options, and the personal options. The client then waits to receive the requested file from the server (step 104). The file that is received has been customized or modified according to the user agent string, and the requested corporate and personal options.
>
> Next, the client checks to see if there is return customization information included in the returned file (step 106). If so, the client reads the return customization information (step

> 108). The client then performs further customization based on the return customization information, along with any other known user preferences and/or current conditions (step 110).
>
> There are many types of client-side customization that may be performed at this point. For example, the server may embed complexity information before various portions of the returned file. The complexity information for a particular portion of the returned file indicates the relative complexity of that particular part of the file. For example, a complexity index may be used, where a larger index indicates more complex data.

However, as can be seen from the above disclosure, there is no teaching or suggestion that is directed to a server, connected with said client through a network, which has means for archiving said image data and low-resolution image data scaled down from said image data and edits said image data. *Fields* clearly discloses that the information is edited in accordance with corporate options and the personal options. There is no teaching or suggestion in *Fields* that is directed to archiving image data and low-resolution image data scaled down from the image data.

As such, Appellant maintains that *Fields* fails to teach or suggest all of the elements recited in claim 46. For at least these reasons, Appellant maintains that claim 46 is not anticipated by *Fields*.

2. ***Fields* fails to teach or suggest "the result of editing being transferred to said server as edit-command information; processed image data being obtained by editing said image data according to said edit-command information at said server; wherein, when giving a command to insert a character image, which represents characters, into said user's image, said image-editing command unit generates character image data representing a character image of the same resolution as said user's image and transfers said character image data and said edit-command information to said server; and said image editing unit obtains said processed image data by inserting said character image into said user's image, based on said edit-command information and said character image data" as required by independent claim 46**

The Examiner relies on col. 6, lines 10-54 and col. 7, lines 1-57 to teach the above elements. However, as noted above with regard to claim 1, there is only one exchange of information between the client and server disclosed in *Fields*. *Fields* fails to teach or suggest the plurality of exchanges between the client and the server as required by claim 46.

Further, there is no teaching or suggestion directed to when giving a command to insert a character image, which represents characters, into said user's image, said image-editing command unit generates character image data representing a character image of the same resolution as said

user's image and transfers said character image data and said edit-command information to said server; and said image editing unit obtains said processed image data by inserting said character image into said user's image, based on said edit-command information and said character image data.

As such, Appellant maintains that *Fields* fails to teach or suggest all of the elements recited in claim 46. For at least these reasons, Appellant maintains that claim 46 is not anticipated by *Fields*.

J. The Rejection Fails to Establish *Prima Facie* Anticipation of Independent Claim 47

Independent claim 47 is directed to an image-editing command unit of an image editing system, equipped with a client having said image-editing command unit for applying a command to edit image data representing a user's image and a server which is connected with said client through a network and has means for archiving said image data and low-resolution image data scaled down from said image data and edits said image data, in which editing information required for editing said image data which contains said low-resolution image data is transferred from said server to said client, an operation of editing said low-resolution image data is performed at said client, the result of editing is transferred to said server as edit-command information, and processed image data is obtained by editing said image data according to said edit-command information at said server, the image-editing command unit comprising means which, when giving a command to insert a character image, which represents characters, into said user's image, generates character image data representing a character image of the same resolution as said user's image and transfers said character image data and said edit-command information to said server.

In support of his rejection of claim 47, the Examiner asserts that *Fields* discloses all of the claim elements citing to col. 6, lines 10-54 and col. 7, lines 1-57. Appellants respectfully disagree with the Examiner's assertions.

1. *Fields* fails to teach or suggest "a server which is connected with said client through a network and has means for archiving said image data and low-resolution image data scaled down from said image data and edits said image data" as required by independent claim 47

The Examiner relies on col. 6, lines 10-54 to teach the above element. However, as can be seen from the above disclosure, there is no teaching or suggestion that is directed to a server, connected with said client through a network, which has means for archiving said image data and

low-resolution image data scaled down from said image data and edits said image data. *Fields* clearly discloses that the information is edited in accordance with corporate options and the personal options. There is no teaching or suggestion in *Fields* that is directed to archiving image data and low-resolution image data scaled down from the image data.

As such, Appellant maintains that *Fields* fails to teach or suggest all of the elements recited in claim 47. For at least these reasons, Appellant maintains that claim 47 is not anticipated by *Fields*.

2. *Fields* fails to teach or suggest "an operation of editing said low-resolution image data is performed at said client, the result of editing is transferred to said server as edit-command information, and processed image data is obtained by editing said image data according to said edit-command information at said server, the image-editing command unit comprising means which, when giving a command to insert a character image, which represents characters, into said user's image, generates character image data representing a character image of the same resolution as said user's image and transfers said character image data and said edit-command information to said server" as required by independent claim 47

The Examiner relies on col. 7, lines 1-57 to teach the above elements. However, as noted above with regard to claim 1, there is only one exchange of information between the client and server disclosed in *Fields*. *Fields* fails to teach or suggest the plurality of exchanges between the client and the server as required by claim 47.

Further, there is no teaching or suggestion directed to when giving a command to insert a character image, which represents characters, into said user's image, generates character image data representing a character image of the same resolution as said user's image and transfers said character image data and said edit-command information to said server.

As such, Appellant maintains that *Fields* fails to teach or suggest all of the elements recited in claim 47. For at least these reasons, Appellant maintains that claim 47 is not anticipated by *Fields*.

K. The Rejection Fails to Establish *Prima Facie* Anticipation of Independent Claim 49

Independent claim 49 is directed to an image-editing command method in an image editing system, equipped with a client having an image-editing command unit for applying a command to edit image data representing a user's image and a server which is connected with said client through

a network and has means for archiving said image data and low-resolution image data scaled down from said image data and edits said image data, in which editing information required for editing said image data which contains said low-resolution image data is transferred from said server to said client, an operation of editing said low-resolution image data is performed at said client, the result of editing is transferred to said server as edit-command information, and processed image data is obtained by editing said image data according to said edit-command information at said server; the image-editing command method comprising the steps of, when giving a command to insert a character image, which represents characters, into said user's image, generating character image data representing a character image of the same resolution as said user's image, and transferring said character image data and said edit-command information to said server.

In support of his rejection of claim 49, the Examiner asserts that *Fields* discloses all of the claim elements citing to col. 6, lines 10-54 and col. 7, lines 1-57. Appellants respectfully disagree with the Examiner's assertions.

1. ***Fields* fails to teach or suggest "a server which is connected with said client through a network and has means for archiving said image data and low-resolution image data scaled down from said image data and edits said image data" as required by independent claim 49**

The Examiner relies on col. 6, lines 10-54 to teach the above element. However, as can be seen from the above disclosure, there is no teaching or suggestion that is directed to a server, connected with said client through a network, which has means for archiving said image data and low-resolution image data scaled down from said image data and edits said image data. *Fields* clearly discloses that the information is edited in accordance with corporate options and the personal options. There is no teaching or suggestion in *Fields* that is directed to archiving image data and low-resolution image data scaled down from the image data.

As such, Appellant maintains that *Fields* fails to teach or suggest all of the elements recited in claim 49. For at least these reasons, Appellant maintains that claim 49 is not anticipated by *Fields*.

2. ***Fields* fails to teach or suggest "an operation of editing said low-resolution image data is performed at said client, the result of editing is transferred to said server as edit-command information, and processed image data is obtained by editing said image data according to said edit-command information at said server; the image-editing command method comprising the steps of, when giving a command to insert a character image, which represents characters, into said user's image, generating character image data representing a character image of the same resolution as said user's image, and transferring said character image data and said edit-command information to said server" as required by independent claim 49**

The Examiner relies on col. 7, lines 1-57 to teach the above elements. However, as noted above with regard to claim 1, there is only one exchange of information between the client and server disclosed in *Fields*. *Fields* fails to teach or suggest the plurality of exchanges between the client and the server as required by claim 49.

Further, there is no teaching or suggestion directed to when giving a command to insert a character image, which represents characters, into said user's image, generating character image data representing a character image of the same resolution as said user's image, and transferring said character image data and said edit-command information to said server.

As such, Appellant maintains that *Fields* fails to teach or suggest all of the elements recited in claim 49. For at least these reasons, Appellant maintains that claim 49 is not anticipated by *Fields*.

L. The Rejection Fails to Establish *Prima Facie* Anticipation of Independent Claim 51

Independent claim 51 is directed to, in a computer readable storage medium, recording a program for causing a computer to carry out an image-editing command method, in an image editing system, equipped with a client having an image-editing command unit for applying a command to edit image data representing a user's image and a server which is connected with said client through a network and has means for archiving said image data and low-resolution image data scaled down from said image data and edits said image data, in which editing information required for editing said image data which contains said low-resolution image data is transferred from said server to said client, an operation of editing said low-resolution image data is performed at said client, the result of

editing is transferred to said server as edit-command information, and processed image data is obtained by editing said image data according to said edit-command information at said server, the computer readable storage medium wherein said program has the procedures of, when giving a command to insert a character image, which represents characters, into said user's image, generating character image data representing a character image of the same resolution as said user's image, and transferring said character image data and said edit-command information to said server.

In support of his rejection of claim 51, the Examiner asserts that *Fields* discloses all of the claim elements citing to col. 6, lines 10-54 and col. 7, lines 1-57. Appellants respectfully disagree with the Examiner's assertions.

1. ***Fields* fails to teach or suggest "a server which is connected with said client through a network and has means for archiving said image data and low-resolution image data scaled down from said image data and edits said image data" as required by independent claim 51**

The Examiner relies on col. 6, lines 10-54 to teach the above element. However, as can be seen from the above disclosure, there is no teaching or suggestion that is directed to a server, connected with said client through a network, which has means for archiving said image data and low-resolution image data scaled down from said image data and edits said image data. *Fields* clearly discloses that the information is edited in accordance with corporate options and the personal options. There is no teaching or suggestion in *Fields* that is directed to archiving image data and low-resolution image data scaled down from the image data.

As such, Appellant maintains that *Fields* fails to teach or suggest all of the elements recited in claim 51. For at least these reasons, Appellant maintains that claim 51 is not anticipated by *Fields*.

2. ***Fields* fails to teach or suggest "an operation of editing said low-resolution image data is performed at said client, the result of editing is transferred to said server as edit-command information, and processed image data is obtained by editing said image data according to said edit-command information at said server, the computer readable storage medium wherein said program has the procedures of, when giving a command to insert a character image, which represents characters, into said user's image, generating character image data representing a character image of the same resolution as said user's image, and transferring said character image data and said edit-command information to said server" as required by independent claim 51**

The Examiner relies on col. 7, lines 1-57 to teach the above elements. However, as noted above with regard to claim 1, there is only one exchange of information between the client and server disclosed in *Fields*. *Fields* fails to teach or suggest the plurality of exchanges between the client and the server as required by claim 51.

Further, there is no teaching or suggestion in *Fields* that is directed to when giving a command to insert a character image, which represents characters, into said user's image, generating character image data representing a character image of the same resolution as said user's image, and transferring said character image data and said edit-command information to said server.

As such, Appellant maintains that *Fields* fails to teach or suggest all of the elements recited in claim 51. For at least these reasons, Appellant maintains that claim 51 is not anticipated by *Fields*.

J. The Rejection Fails to Establish *Prima Facie* Anticipation of Dependent Claims 8-19 and 33-45

Dependent claims 8-19 and 33-45 depend, either directly or indirectly from claim 7. Appellant respectfully submits that claims 8-19 and 33-45 are allowable for the reasons set forth above with regard to claim 7 at least based upon their dependency on claim 7. Appellant further submits that dependent claims 8-19 and 33-45 are separately patentable and offers the following additional argument for the invention of claims 8-19 and 33-45.

The rejection of claims 8-19 and 33-45 asserts that *Fields* teaches the incremental features as cited therein. Appellant submits, however, that the rejection's reliance on *Fields* as recited as allegedly teaching this incremental feature fails to make up for the deficiencies of the rejection

applied to claim 7. Thus the Examiner fails to establish *prima facie* anticipation of dependent claims 8-19 and 33-45.

K. The Rejection Fails to Establish *Prima Facie* Anticipation of Dependent Claims 21-32

Dependent claims 21-32 depend, either directly or indirectly from claim 20. Appellant respectfully submits that claims 21-32 are allowable for the reasons set forth above with regard to claim 20 at least based upon their dependency on claim 20. Appellant further submits that dependent claims 21-32 are separately patentable and offers the following additional argument for the invention of claims 21-32.

The rejection of claims 21-32 asserts that *Fields* teaches the incremental features as cited therein. Appellant submits, however, that the rejection's reliance on *Fields* as recited as allegedly teaching this incremental feature fails to make up for the deficiencies of the rejection applied to claim 20. Thus the Examiner fails to establish *prima facie* anticipation of dependent claims 21-32.

L. The Rejection Fails to Establish *Prima Facie* Anticipation of Dependent Claim 48

Dependent claim 48 depends, directly from claim 47. Appellant respectfully submits that claim 48 is allowable for the reasons set forth above with regard to claim 47 at least based upon its dependency on claim 47. Appellant further submits that dependent claim 48 is separately patentable and offers the following additional argument for the invention of claim 48.

The rejection of claim 48 asserts that *Fields* teaches the incremental features as cited therein. Appellant submits, however, that the rejection's reliance on *Fields* as recited as allegedly teaching this incremental feature fails to make up for the deficiencies of the rejection applied to claim 47. Thus the Examiner fails to establish *prima facie* anticipation of dependent claim 48.

M. The Rejection Fails to Establish *Prima Facie* Anticipation of Dependent Claims 50 and 52

Dependent claims 50 and 52 depend, either directly or indirectly from claim 49. Appellant respectfully submits that claims 50 and 52 are allowable for the reasons set forth above with regard to claim 49 at least based upon their dependency on claim 49. Appellant further submits that

dependent claims 50 and 52are separately patentable and offers the following additional argument for the invention of claims 50 and 52.

The rejection of claims 50 and 52 asserts that *Fields* teaches the incremental features as cited therein. Appellant submits, however, that the rejection's reliance on *Fields* as recited as allegedly teaching this incremental feature fails to make up for the deficiencies of the rejection applied to claim 49. Thus the Examiner fails to establish *prima facie* anticipation of dependent claims 50 and 52.

VIII. CONCLUSION

The withdrawal of the outstanding rejections and the allowance of claims 1-52 is earnestly solicited.

Respectfully submitted,

By _____
Marc S. Weiner
Registration No.: 32,181
BIRCH, STEWART, KOLASCH & BIRCH, LLP
8110 Gatehouse Road
Suite 100 East
P.O. Box 747
Falls Church, Virginia 22040-0747
(703) 205-8000
Attorney for Appellant

IX. CLAIMS APPENDIX

1. (Previously Presented) An image editing method that is performed in an image editing system equipped with a client, which has an edit-command unit for applying a command to edit image data, and an image server, connected with said client through a network, which has an editing unit for obtaining processed image data by editing said image data in response to the edit command from said edit-command unit, said image editing method comprising:

a first step of accepting an edit-start command and, in response to said edit-start command, commanding said image server to transfer editing data, having at least one editing object, which contains said image data, at said edit-command unit, and of transferring said editing data to said client at said image server;

a second step of querying said image server about one editing object for obtaining said processed image data in accordance with said editing data, at said edit-command unit;

a third step of transferring editing information, which represents said one editing object corresponding to said inquiry, to said client, at said editing unit;

a fourth step of generating edit-command information which represents a command to edit said editing object, in accordance with said editing information and also transferring said edit-command information to said image server, at said edit-command unit;

a fifth step of obtaining intermediate processed image data by applying an editing process to said editing data in accordance with said edit-command information and also transferring said intermediate processed image data to said client, at said editing unit; and

a sixth step of repeating said second through the fifth steps, until said edit-command information is transferred for an editing object desired and said processed image data is obtained.

2. (Previously Presented) An image editing system comprising:

a client having an edit-command unit for applying a command to edit image data;

an image server, connected with said client through a network, which has an editing unit for obtaining processed image data by editing said image data in response to the edit command from said edit-command unit;

said edit-command unit having first means for accepting an edit-start command and, in response to said edit-start command, commanding said image server to transfer editing data, having at least one editing object, which contains said image data; second means for querying said image server about one editing object for obtaining said processed image data, based on said editing data transferred from said image server in accordance with said command to transfer said editing data; and third means for generating edit-command information which represents a command to edit said editing object, based on said editing information transferred from said image server in accordance with said inquiry about said editing object, and for transferring said edit-command information to said image server;

said editing unit having first means for transferring said editing data to said client in response to said command to transfer said editing data; second means for transferring editing information, which represents an editing object corresponding to said inquiry, to said client; and third means for obtaining intermediate processed image data by applying an editing process to said editing data, based on said edit-command information, and for transferring said intermediate processed image data to said client; and

means for repeatedly carrying out the steps carried out in the second and third means of said edit-command unit and the first, second, and third means of said editing unit, until said edit-

command information is transferred for an editing object desired and said processed image data is obtained.

3. (Original) A computer readable storage medium recording a program for causing a computer to carry out the image editing method as set forth in claim 1, wherein said program has a first procedure of accepting an edit-start command and, in response to said edit-start command, commanding said image server to transfer editing data, having at least one editing object, which contains said image data;

a second procedure of querying said image server about one editing object for obtaining said processed image data, based on said editing data transferred from said image server in accordance with said command to transfer said editing data;

a third procedure of generating edit-command information which represents a command to edit said editing object, based on said editing information transferred from said image server in accordance with said inquiry about said editing object, and of transferring said edit-command information to said image server; and

a fourth procedure of repeating said second and third procedures, until said edit-command information is transferred for an editing object desired and said processed image data is obtained.

4. (Original) A computer readable storage medium recording a program for causing a computer to carry out the image editing method as set forth in claim 1, wherein said program has a first procedure of transferring said editing data to said client in response to said command to transfer said editing data;

a second procedure of transferring editing information, which represents an editing object corresponding to said inquiry, to said client;

a third procedure of obtaining intermediate processed image data by applying an editing process to said editing data, based on said edit-command information, and of transferring said intermediate processed image data to said client; and

a fourth procedure of repeating said first, second, and third procedures, until said edit-command information is transferred for an editing object desired and said processed image data is obtained.

5. (Previously Presented) An edit-command unit in an image editing system equipped with a client, which has said edit-command unit for applying a command to edit image data, and an image server, connected with said client through a network, which has an editing unit for obtaining processed image data by editing said image data in response to the edit command from said edit-command unit, said edit-command unit comprising:

first means for accepting an edit-start command and, in response to said edit-start command, commanding said image server to transfer editing data, having at least one editing object, which contains said image data;

second means for querying said image server about one editing object for obtaining said processed image data, based on said editing data transferred from said image server in accordance with said command to transfer said editing data;

third means for generating edit-command information which represents a command to edit said editing object, based on said editing information transferred from said image server in accordance with said inquiry about said editing object, and for transferring said edit-command information to said image server; and

fourth means for repeatedly carrying out the steps carried out in said second and third means, until said edit-command information is transferred for an editing object desired and said processed image data is obtained.

6. (Previously Presented) An editing unit in an image editing system equipped with a client, which has an edit-command unit for giving a command to edit image data, and an image server, connected with said client through a network, which has said editing unit for obtaining processed image data by editing said image data in response to the edit command from said edit-command unit, said editing unit comprising:

first means for transferring said editing data to said client in response to said command to transfer said editing data;

second means for transferring editing information, which represents an editing object corresponding to said inquiry, to said client;

third means for obtaining intermediate processed image data by applying an editing process to said editing data, based on said edit-command information, and for transferring said intermediate processed image data to said client; and

fourth means for repeatedly carrying out the steps carried out in said first, second, and third means, until said edit-command information is transferred for an editing object desired and said processed image data is obtained.

7. (Previously Presented) An image editing method that is performed in an image editing system equipped with a client, which has an edit-command unit for giving a command to edit image data, and an image server, connected with said client through a network, which has an editing unit for obtaining processed image data by performing an editing process on said image data in response

to the edit command from said edit-command unit and transfers predetermined image data related to said image data to said client, said image editing method comprising the steps of:

generating low-volume data smaller in data amount than said predetermined image data; and

transferring said low-volume data to said client.

8. (Original) The image editing method as set forth in claim 7, wherein said predetermined image data is any one among image data before said editing process is applied, image data subjected to an editing process up to an intermediate stage, and said processed image data.

9. (Original) The image editing method as set forth in claim 7, wherein said predetermined image data is transferred to said client, following said low-volume data.

10. (Original) The image editing method as set forth in claim 8, wherein said predetermined image data is transferred to said client, following said low-volume data.

11. (Original) The image editing method as set forth in claim 7, wherein the data amount of said low-volume data is varied according to a loaded state of said network.

12. (Original) The image editing method as set forth in claim 8, wherein the data amount of said low-volume data is varied according to a loaded state of said network.

13. (Original) The image editing method as set forth in claim 9, wherein the data amount of said low-volume data is varied according to a loaded state of said network.

14. (Original) The image editing method as set forth in claim 7, wherein said low-volume data is composed of a plurality of data reduced in stages in data amount and is transferred to said client from the data smaller in data amount.

15. (Original) The image editing method as set forth in claim 8, wherein said low-volume data is composed of a plurality of data reduced in stages in data amount and is transferred to said client from the data smaller in data amount.

16. (Original) The image editing method as set forth in claim 9, wherein said low-volume data is composed of a plurality of data reduced in stages in data amount and is transferred to said client from the data smaller in data amount.

17. (Original) The image editing method as set forth in claim 11, wherein said low-volume data is composed of a plurality of data reduced in stages in data amount and is transferred to said client from the data smaller in data amount.

18. (Original) The image editing method as set forth in claim 14, wherein transfer of said low-volume data is suspended in response to a command from said client.

19. (Original) The image editing method as set forth in claim 18, wherein transfer of said low-volume data is restarted in response to a command from said client.

20. (Previously Presented) An image editing system comprising:

a client having an edit-command unit for giving a command to edit image data;

an image server, connected with said client through a network, which has an editing unit for obtaining processed image data by performing an editing process on said image data in response to the edit command from said edit-command unit and transfers predetermined image data related to said image data to said client;

wherein said image server has means for generating low-volume data smaller in data amount than said predetermined image data, and transfers said low-volume data to said client.

21. (Original) The image editing system as set forth in claim 14, wherein said predetermined image data is any one among image data before said editing process is applied, image data subjected to an editing process up to an intermediate stage, and said processed image data.

22. (Original) The image editing system as set forth in claim 20, wherein said image server is further equipped with means for transferring said predetermined image data to said client, following said low-volume data.

23. (Original) The image editing system as set forth in claim 21, wherein said image server is further equipped with means for transferring said predetermined image data to said client, following said low-volume data.

24. (Original) The image editing system as set forth in claim 20, wherein said image server is further equipped with means for varying the data amount of said low-volume data according to a loaded state of said network.

25. (Original) The image editing system as set forth in claim 21, wherein said image server is further equipped with means for varying the data amount of said low-volume data according to a loaded state of said network.

26. (Original) The image editing system as set forth in claim 22, wherein said image server is further equipped with means for varying the data amount of said low-volume data according to a loaded state of said network.

27. (Original) The image editing system as set forth in claim 20, wherein said means for generating low-volume data is means for generating said low-volume data so that it is composed of a plurality of data reduced in stages in data amount, and transfers said low-volume data to said client in sequence from the data having a smaller data amount.

28. (Original) The image editing system as set forth in claim 21, wherein said means for generating low-volume data is means for generating said low-volume data so that it is composed of a plurality of data reduced in stages in data amount, and transfers said low-volume data to said client in sequence from the data having a smaller data amount.

29. (Original) The image editing system as set forth in claim 22, wherein said means for generating low-volume data is means for generating said low-volume data so that it is composed of a plurality of data reduced in stages in data amount, and transfers said low-volume data to said client in sequence from the data having a smaller data amount.

30. (Original) The image editing system as set forth in claim 24, wherein said means for generating low-volume data is means for generating said low-volume data so that it is composed of a plurality of data reduced in stages in data amount, and transfers said low-volume data to said client in sequence from the data smaller in data amount.

31. (Original) The image editing system as set forth in claim 27, wherein said image server is further equipped with means for suspending transfer of said low-volume data in response to a command from said client.

32. (Original) The image editing system as set forth in claim 31, wherein said image server is further equipped with means for restarting transfer of said low-volume data in response to a command from said client.

33. (Original) A computer readable storage medium recording a program for causing a computer to carry out the image editing method as set forth in claim 7, wherein said program has a procedure of generating low-volume data smaller in data amount than said predetermined image data; and

a procedure of transfers said low-volume data to said client.

34. (Original) The computer readable storage medium as set forth in claim 33, wherein said predetermined image data is any one among image data before said editing process is applied, image data subjected to an editing process up to an intermediate stage, and said processed image data.

35. (Original) The computer readable storage medium as set forth in claim 33, wherein said program further has a procedure of transferring said predetermined image data to said client, following said low-volume data.

36. (Original) The computer readable storage medium as set forth in claim 34, wherein said program further has a procedure of transferring said predetermined image data to said client, following said low-volume data.

37. (Original) The computer readable storage medium as set forth in claim 33, wherein said program further has a procedure of varying the data amount of said low-volume data according to a loaded state of said network.

38. (Original) The computer readable storage medium as set forth in claim 34, wherein said program further has a procedure of varying the data amount of said low-volume data according to a loaded state of said network.

39. (Original) The computer readable storage medium as set forth in claim 35, wherein said program further has a procedure of varying the data amount of said low-volume data according to a loaded state of said network.

40. (Original) The computer readable storage medium as set forth in claim 33, wherein said low-volume data is composed of a plurality of data reduced in stages in data amount, and said procedure of transferring low-volume data is a procedure of transferring said low-volume data to said client in sequence from the data having a smaller data amount.

41. (Original) The computer readable storage medium as set forth in claim 34, wherein said low-volume data is composed of a plurality of data reduced in stages in data amount, and said procedure of transferring low-volume data is a procedure of transferring said low-volume data to said client in sequence from the data smaller in data amount.

42. (Original) The computer readable storage medium as set forth in claim 35, wherein said low-volume data is composed of a plurality of data reduced in stages in data amount, and said procedure of transferring low-volume data is a procedure of transferring said low-volume data to said client in sequence from the data smaller in data amount.

43. (Original) The computer readable storage medium as set forth in claim 37, wherein said low-volume data is composed of a plurality of data reduced in stages in data amount, and said procedure of transferring low-volume data is a procedure of transferring said low-volume data to said client in sequence from the data having a smaller data amount.

44. (Original) The computer readable storage medium as set forth in claim 40, wherein said program further has a procedure of suspending transfer of said low-volume data in response to a command from said client.

45. (Original) The computer readable storage medium as set forth in claim 44, wherein said program further has a procedure of restarting transfer of said low-volume data in response to a command from said client.

46. (Previously Presented) An image editing system comprising:

a client having an image-editing command unit for applying a command to edit image data representing a user's image; and

a server, connected with said client through a network, which has means for archiving said image data and low-resolution image data scaled down from said image data and edits said image data;

editing information required for editing said image data which contains said low-resolution image data being transferred from said server to said client;

an operation of editing said low-resolution image data being performed at said client;

the result of editing being transferred to said server as edit-command information;

processed image data being obtained by editing said image data according to said edit-command information at said server;

wherein, when giving a command to insert a character image, which represents characters, into said user's image, said image-editing command unit generates character image data representing a character image of the same resolution as said user's image and transfers said character image data and said edit-command information to said server; and

said image editing unit obtains said processed image data by inserting said character image into said user's image, based on said edit-command information and said character image data.

47 (Previously Presented) An image-editing command unit of an image editing system, equipped with a client having said image-editing command unit for applying a command to edit image data representing a user's image and a server which is connected with said client through a network and has means for archiving said image data and low-resolution image data scaled down from said image data and edits said image data, in which editing information required for editing said image data which contains said low-resolution image data is transferred from said server to said client, an operation of editing said low-resolution image data is performed at said client, the result of editing is transferred to said server as edit-command information, and processed image data is obtained by editing said image data according to said edit-command information at said server,

the image-editing command unit comprising means which, when giving a command to insert a character image, which represents characters, into said user's image, generates character image data representing a character image of the same resolution as said user's image and transfers said character image data and said edit-command information to said server.

48. (Original) An image editing unit for editing image data in accordance with the edit-command information obtained in the image-editing command unit as set forth in claim 47, said image editing unit comprising means for obtaining processed image data by inserting a character image into a user's image, based on said edit-command information and character image data.

49. (Previously Presented) An image-editing command method in an image editing system, equipped with a client having an image-editing command unit for applying a command to edit image data representing a user's image and a server which is connected with said client through a network and has means for archiving said image data and low-resolution image data scaled down from said image data and edits said image data, in which editing information required for editing said image data which contains said low-resolution image data is transferred from said server to said client, an operation of editing said low-resolution image data is performed at said client, the result of editing is transferred to said server as edit-command information, and processed image data is obtained by editing said image data according to said edit-command information at said server;

the image-editing command method comprising the steps of, when giving a command to insert a character image, which represents characters, into said user's image, generating character image data representing a character image of the same resolution as said user's image, and transferring said character image data and said edit-command information to said server.

50. (Original) An image editing method of editing image data in accordance with the edit-command information obtained in the image-editing command method as set forth in claim 49, said image editing method comprising the step of obtaining processed image data by inserting a character image into a user's image in accordance with said edit-command information and character image data.

51. (Previously Presented) In a computer readable storage medium, recording a program for causing a computer to carry out an image-editing command method, in an image editing system, equipped with a client having an image-editing command unit for applying a command to edit image data representing a user's image and a server which is connected with said client through a network and has means for archiving said image data and low-resolution image data scaled down from said image data and edits said image data, in which editing information required for editing said image data which contains said low-resolution image data is transferred from said server to said client, an operation of editing said low-resolution image data is performed at said client, the result of editing is transferred to said server as edit-command information, and processed image data is obtained by editing said image data according to said edit-command information at said server,

the computer readable storage medium wherein said program has the procedures of, when giving a command to insert a character image, which represents characters, into said user's image, generating character image data representing a character image of the same resolution as said user's image, and transferring said character image data and said edit-command information to said server.

52. (Original) A computer readable storage medium recording a program for causing a computer to carry out a method of editing image data in accordance with the edit-command information obtained in the image-editing command method as set forth in claim 49, wherein said program has a procedure of obtaining processed image data by inserting a character image into a user's image in accordance with said edit-command information and character image data.

X. EVIDENCE APPENDIX

No evidence has been submitted under 37 C.F.R 1.130, 1.131 or 1.132. No other evidence has been entered by the Examiner and relied upon in this appeal.

XI. RELATED PROCEEDINGS APPENDIX

There are no related proceedings.